AMENDMENTS TO THE CLAIMS

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1-39. (Previously Cancelled)

40. (Currently Amended) A configuration management system comprising:

a memory component:

a standardized configuration store that stores persisted information associated with settings for each of a plurality of instances of an application onto the memory component according to a uniform semantics scheme, the storage of persisted information for each ~~any one~~ of the plurality of instances isolated from persisted information for all ~~any~~ of the remaining plurality of instances, the persisted information being isolated according to a unique namespace for each instance; and

a configuration service component that manages access to the standardized configuration store and converts information associated with the application into the persisted information associated with each of the plurality of instances of the application;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

41. (Previously Presented) The system of claim 40, the information associated with an application is at least one of configuration information or dependency information.

42. (Previously Presented) The system of claim 40, wherein the configuration service component receives a manifest associated with the application, the manifest comprising at least one of configuration and dependency information associated with the application, and the configuration service component converts and stores at least some of the manifest information in

the configuration store.

43. (Previously Presented) The system of claim 42, wherein the manifest is based, at least in part, upon a schema.

44. (Previously Presented) The system of claim 43, wherein the schema is XML-based.

45. (Previously Presented) The system of claim 42, wherein the manifest employing at least one of strong typing, validation, and assertions.

46. (Previously Presented) The system of claim 42, wherein the configuration service component compiles at least one of manifest information into a namespace, the configuration service component providing access to the namespace.

47. (Previously Presented) The system of claim 40, further comprising a configuration management engine that identifies configuration information within the persisted information and facilitates management of at least a portion of the configuration information.

48. (Previously Presented) The system of claim 40, the configuration service component facilitating access to a legacy store.

49. (Previously Presented) The system of claim 48, the legacy store comprising a registry.

50. (Previously Presented) The system of claim 40, the configuration service component facilitating at least one management service.

51. (Previously Presented) The system of claim 50, the management service comprising at least one of a group policy component and a roaming component.

52. (Previously Presented) The system of claim 50, the management service facilitating at least one of install, usage, servicing, uninstall, roaming, migration, setup, provisioning, policy, backup and/or restore.

53. (Previously Presented) The system of claim 40, further comprising an assertion engine that facilitates administration of a validation rule by the configuration service component.

54. (Previously Presented) The system of claim 40, further comprising a notification handler that provides information associated with a configuration change of the application to at least one of the application and another application.

55. (Previously Presented) The system of claim 40, further comprising a legacy handler that facilitates synchronization of the system with a legacy store.

56. (Previously Presented) The system of claim 55, the legacy store comprising a registry.

57. (Previously Presented) The system of claim 40, wherein the configuration service component facilitates transacted commits for saving related changes together in the configuration store.

58. (Previously Presented) The system of claim 40, wherein the configuration service component employs at least one of ACL-based security and role-based security are provided at per-setting granularity.

59. (Previously Presented) The system of claim 40, wherein the configuration service component facilitates change logs and history.

60. (Previously Presented) The system of claim 40, wherein the configuration store comprises a joint engine technology database that stores a settings namespace.

61. (Previously Presented) The system of claim 60, wherein a namespace comprises metadata on settings comprising types, attributes, and user context, the namespace further comprising instance values of the settings.

62. (Previously Presented) The system of claim 61, wherein at least one of the metadata on the settings and instance values of the settings is stored for each user context.

63. (Previously Presented) The system of claim 40, wherein at least one of URI and Xpath can access a setting within a namespace as well as in between namespaces.

64. (Currently Amended) A configuration management system comprising:

a local cache that at least temporarily stores changes to persisted information associated with settings for an application; and

a configuration management engine that facilitates communication of the changed persisted information stored in the local cache to a configuration service component, the configuration management engine facilitating an isolation of the changed persisted information at least until a notification is received that the changed persisted information has been committed, the persisted information being isolated according to a unique namespace for each instance of the application;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

65. (Previously Presented) The system of claim 64, the persisted information comprising at least a standardized representation of configuration information.

66. (Previously Presented) The system of claim 65, the configuration information comprises at least information other than dependency information.

67. (Currently Amended) A method for facilitating configuration management comprising: receiving a manifest associated with an application, the manifest comprising at least configuration information and dependency information associated with a plurality of instances of the application;

registering the manifest;

processing the manifest to generate persisted information associated with settings for each of the plurality of instances of the application from at least one of the configuration information or the dependency information for each of the plurality of instances; and

storing at least some of the persisted information in a standardized configuration store according to a uniform semantics scheme, the persisted information for each ~~any one~~ of the plurality of instances isolated from persisted information for all ~~any~~ of the remaining plurality of instances, the persisted information being isolated according to a unique namespace for each instance;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

68. (Previously Presented) The method of claim 67, further comprising compiling at least a portion of the persisted information into a namespace.

69. (Currently Amended) A computer readable medium having stored thereon computer executable instructions for:

receiving a manifest associated with an application, the manifest comprising at least configuration information and dependency information associated with a plurality of instances of the application;

registering the manifest;

processing the manifest to generate persisted information associated with settings for each of the plurality of instances of the application from at least one of the configuration information or the dependency information for each of the plurality of instances; and

storing at least some of the persisted information in a standardized configuration store according to a uniform semantics scheme, the persisted information for each ~~any one~~ of the plurality of instances isolated from persisted information for all ~~any~~ of the remaining plurality of instances, the persisted information being isolated according to a unique namespace for each instance;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

70. (Currently Amended) A method of facilitating configuration management comprising:

providing a manifest, the manifest associated with configuration information and dependency information of a first application;

processing the manifest to generate persistent information associated with settings for the first application;

storing the persistent information according to a uniform semantics scheme so as to isolate information associated with each of a plurality of instances of the first application, the persisted information being isolated according to a unique namespace for each instance; and

accessing a configuration setting of the first application within the persistent information;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

71. (Previously Presented) The method of claim 70 further comprising at least one of the following acts:

identifying settings in a namespace associated with the first application;

defining a name, a type, a description and default value for a setting;

defining other metadata for the setting;

providing a validation rule for the setting;

indicating service applicability for the setting; and,

identifying a dependency using an assertion expression.

72. (Previously Presented) The method of claim 71, further comprising at least one of the following acts:

accessing a setting associated with the first application; and,

accessing a setting associated with a second application.

73. (Currently Amended) A computer readable medium having stored thereon computer executable instructions for:

providing a manifest, the manifest associated with configuration information and dependency information of a first application;

processing the manifest to generate persistent information associated with settings for the first application;

storing the persistent information according to a uniform semantics scheme so as to isolate information associated with each of a plurality of instances of the first application, the persisted information being isolated according to a unique namespace for each instance of the application; and

accessing a configuration setting of the first application within the persistent information;

wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

74. (Currently Amended) A computer readable medium storing computer executable components of a configuration management system comprising:

a configuration service component that manages access to a standardized configuration store, the configuration service component comprising an assertion engine component and a legacy handler component, wherein the standardized configuration store stores persisted information associated with settings for each of a plurality of instances of an application according to a uniform semantics scheme, the persisted information for each ~~any one~~ of the plurality of instances isolated from persisted information for all ~~any~~ of the remaining plurality of instances, the persisted information being isolated according to a unique namespace for each instance, the assertion engine component facilitates administration of a validation rule by the configuration service component, and the legacy handler component facilitates synchronization with a legacy store including a registry, wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application.

75. (Currently Amended) A configuration management system comprising:

means for representing configuration information associated with settings for each of a plurality of instances of an application as persisted information, the persisted information comprising a standardized structure for all applications and configuration information;

means for storing the persisted information associated with an each of the plurality of instances of the application according to a uniform semantics scheme, the means for storing including means for storing persisted information for each ~~any one~~ of the plurality of instances in isolation from persisted information for all ~~any~~ of the remaining plurality of instances, the persisted information being isolated according to a unique namespace for each instance, wherein each unique namespace for each instance of the application is derived from each of the following: a name of the application, a version of the instance of the application, a language of the instance of the application, a deployment id for the instance of the application, a process architecture for the instance of the application and a public key token of the instance of the application;

means for managing access to the means for storing persisted information;

means for facilitating administration of a validation rule; and

means for synchronizing the means for storing persisted information with a legacy store including a registry.